Pattern Recognition Letters 109 (2018) 55–64

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Semantic segmentation of RGBD images based on deep depth regression[☆]

Yanrong Guo[*], Tao Chen

School of Computer and Information, Hefei University of Technology, Hefei 230009, China

ARTICLE INFO

Article history:
Available online 24 August 2017

Keywords:
Deep depth regression
RGBD semantic segmentation
Convolutional neural network
Fully convolutional network

ABSTRACT

Depth information has been discovered to improve the performance of computer vision tasks, such as semantic segmentation and object recognition. However, careful acquisition of depth data needs highly developed depth sensors which are expensive. As a classic computer vision task, depth estimation from a single image has obtained promising results based on supervised learning methods. In this paper, we investigate the extension of color images with corresponding deep-regressed depth images in boosting the performance of semantic segmentation. Furthermore, the usage of combining color channels with the estimated depth or the ground truth depth channel is compared. Specifically, there are two stages in our work. Firstly, we adopt the framework of convolutional neural networks (CNN) for the depth estimation by combing the global depth network and the depth gradient network. After refining based on these two networks, the depth image map can be estimated in a deep-regressed manner. Secondly, after augmenting the color images with the predicted depth images, fully convolutional networks (FCN) are further used to implement the pixel-level semantic labeling. In the experiments, we employ two popular RGBD datasets, i.e., SUNRGBD and NYUDv2, for 37 and 40-class semantic segmentation, respectively. By comparing with the ground truth depth images, experimental results demonstrate that the networks trained on the estimated depth images can achieve comparable performance on facilitating the accuracy of semantic segmentation task.

1. Introduction

To assist the scene understanding in computer vision, depth estimation and semantic segmentation are two important research fields in terms of dense predictions [7,17,20]. Machining learning based algorithms [13,26,29,32,34] can be used for solving computer vision tasks from low-level to high-level. As a middle-level task, high-quality depth prediction contributes to the description of geometric information with a scene, which can be used to the tasks such as object detection [2], scene labeling [16] and image enhancement [12]. As a high-level task, semantic segmentation assigns a semantic label to each image pixel based on the scene categories, which helps the tasks such as semantic parsing [3], robot path planning [27].

In this paper, we first adopt the deep convolutional neural network (CNN) [14] for estimating depth from single RGB images. Then after combing the estimated depth with original RGB images, fully convolutional network [19] is further used to labeling each object in the scene. One common observation has already been established that learning across different modalities [4,24,31] can be effective to boost performance. Specifically, with the help of rich information about the object relationships provided in the depth images, the accuracy of semantic segmentation can be improved compared to the segmentation performance with only RGB input data [2]. Apart from this finding, another interesting comparison is between the labeling performances of using predicted depth or the ground truth depth acquired from the depth sensor. In this way, the practical advantage of depth estimation and their methods can be evaluated. Besides, the performance gap between actual depth and the estimated one can help to motivate the improvement of depth estimation methods. Based on the above considerations, our contributions in this paper can be concluded as follows.

- Combining the tasks of depth estimation and semantic segmentation into a unified framework for automatic generating depth image and then applying to formatting RGBD data.
- We discuss the improvement of semantic labeling through inferring depth information and its performance difference compared with using ground truth.

[☆] This work was supported by National Natural Science Foundation of China (Grant No. 61702156).

[*] Corresponding author.
E-mail address: gyr0716@gmail.com (Y. Guo).

http://dx.doi.org/10.1016/j.patrec.2017.08.026
0167-8655/© 2017 Elsevier B.V. All rights reserved.

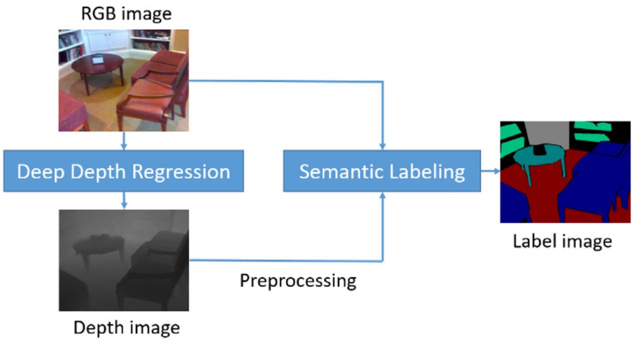


Fig. 1. The overall framework of our RGBD semantic segmentation method.

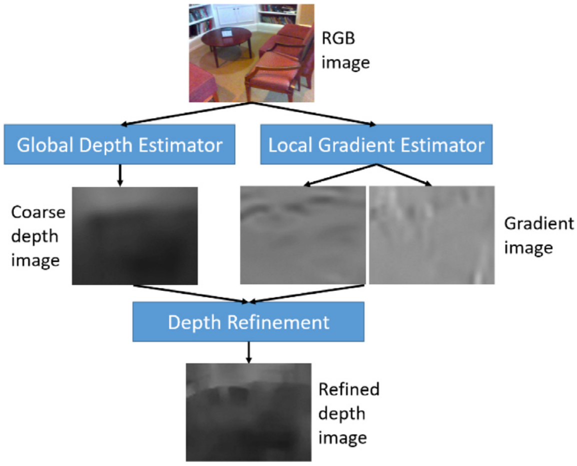


Fig. 2. General scheme of deep-regressed depth estimation model.

For the rest of the paper, Section 2 introduces the related work for depth estimation and semantic segmentation. A detailed description of our work is given in Section 3. Section 4 evaluates the comparison methods on two datasets: NYUDv2 and SUNRGBD. Finally, Section 5 concludes the paper.

2. Related work

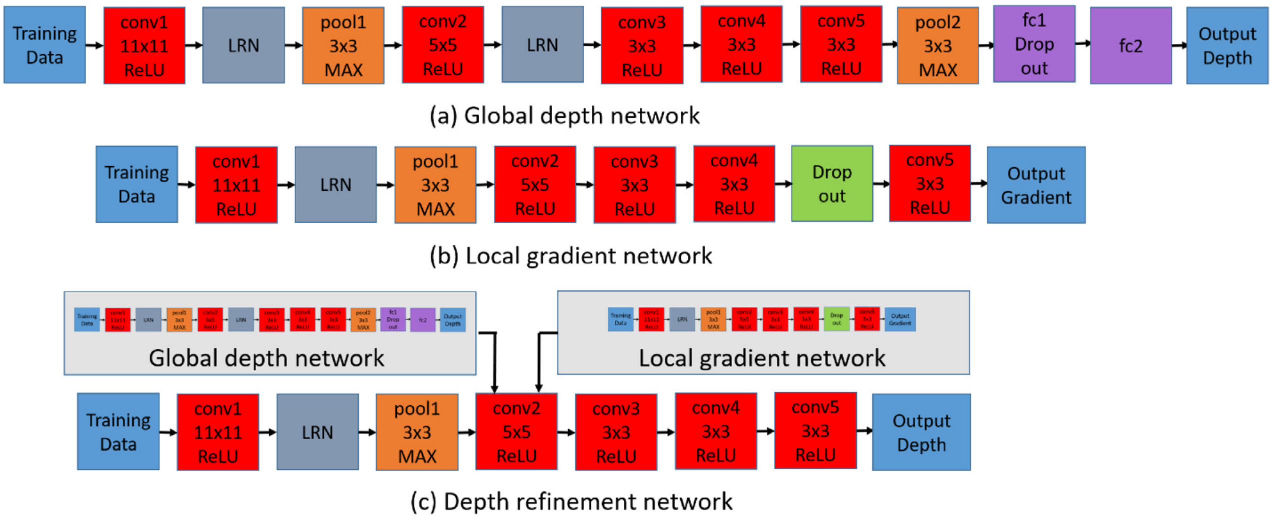
2.1. Depth estimation

Learning based algorithms are among the most effective methods for depth estimation task. According to the different types of learning strategies, depth estimation methods can be divided into two categories, i.e., supervised learning and unsupervised learning methods. According to the different types of inputs, depth information can be learned from a single image, stereo images or motion sequences.

Graph learning methods [5,6,28,30,33] are extensively used in different tasks. For the depth estimation from a single image, supervised methods such as MRF [22], neural regression forest [21], deep convolutional neural networks [14] have obtained very promising results. Eigen et al. [8] proposed the coarse-to-fine deep network stacks, which archived good performance on the NYUDv2 and KITTI datasets. Liu et al. [18] built a deep convolutional neural field model for jointly learning the deep CNN and continuous CRF. Since supervised learning methods require a large amount of ground truth depth images for training stage, which is not always satisfied for some datasets, unsupervised learning methods are proposed to refrain from this limitation. Godard et al. [10] learned the disparities between left and right images of binocular stereo pairs by constraining the consistency for training loss. Garg et al. [9] learned an unsupervised CNN for depth estimation of single images while training on the stereo pairs based on the reconstruction loss.

2.2. Semantic segmentation

Semantic labeling for both indoor scenes and out scenes is a challenging task due to the large variation between and within the complicated objects, as well as the existence of occlusions. To increase the discriminative power of the segmentation methods, robust and effective feature representation is highly desired. Besides, with the help of geometric information provided in the depth images, RGBD data can achieve better performance than using the RGB data alone in terms of segmentation accuracy. Gupta et al. [11] proposed a depth-aware hierarchical segmentation method based on the gPb-ucm architecture, which made use of the geometric contour features from depth images. Apart from the hand-designed features, deep feature representation learned from the deep neural networks attracts more and more attention. Long et al. [19] built a fully convolutional network (FCN) which combined the appearance cues from coarse to fine layers. This method can be applied to the scene labeling of both RGB data and RGBD data. Different from FCN, Bansal et al. [1] presented a sampling strategy training to improve the efficiency and effectiveness. Li et al.

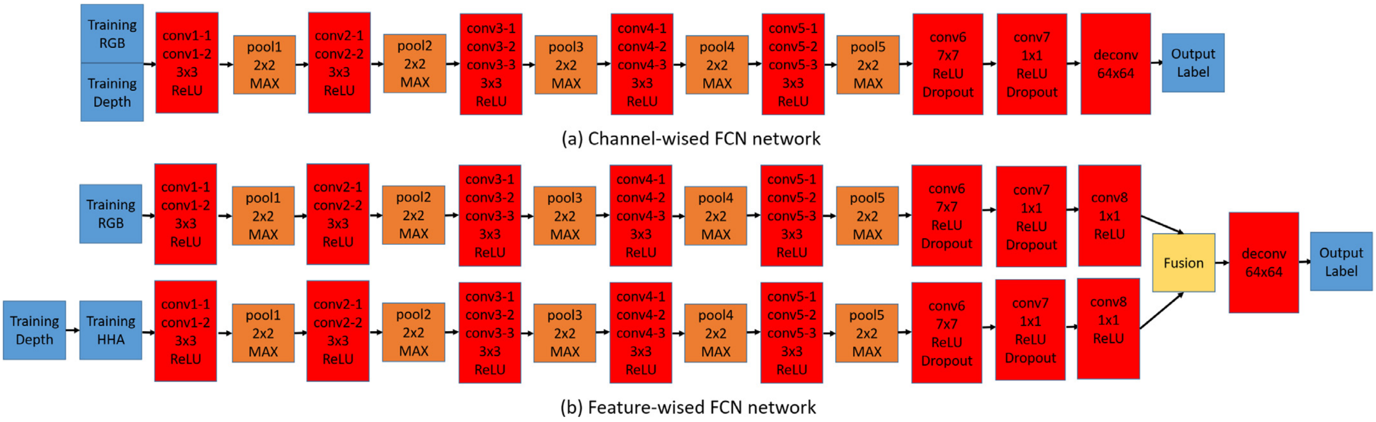


(a) Global depth network

(b) Local gradient network

(c) Depth refinement network

Fig. 3. CNN based deep-regressed depth model.



(a) Channel-wised FCN network

(b) Feature-wised FCN network

Fig. 4. FCN32 based RGBD semantic segmentation model.

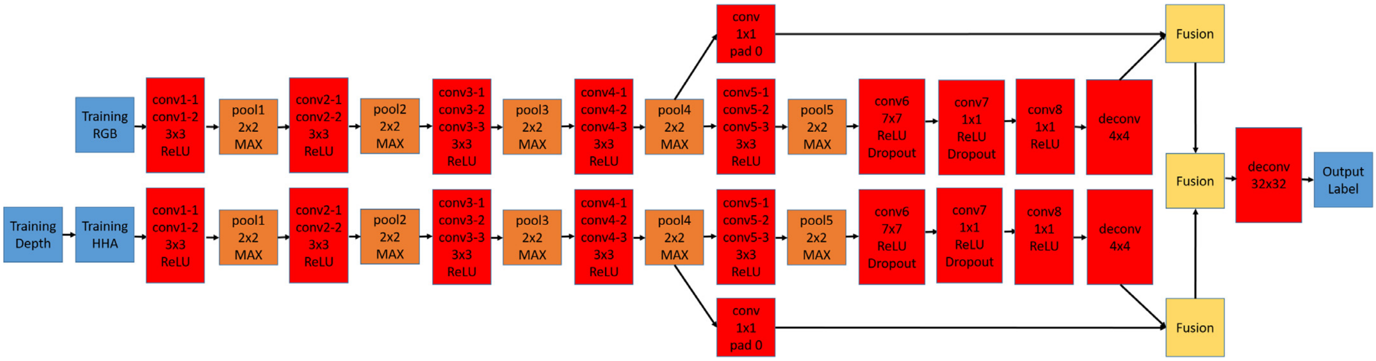


Fig. 5. FCN16 based RGBD semantic segmentation model.

Table 1
Parameters used for building network structures.

	GDN	LGN	DRN	C-FCN	F-FCN
Learning rate	5×10^{-4}	2.5×10^{-4}	2.5×10^{-5}	1×10^{-10}	1×10^{-10}
Momentum weight	0.9	0.9	0.9	0.99	0.99
Decay weight	5×10^{-3}	5×10^{-3}	5×10^{-3}	5×10^{-4}	5×10^{-4}
Batch size	32	32	16	1	1

[16] proposed a deep CNN for mining the contextual information embedded in both the RGB and depth images.

3. Method

In this section, we introduce the proposed framework of a two-stage learning method for semantic segmentation based on estimated depth information. Generally, we first train a CNN for single-image depth estimation and then build an FCN model based on the image pair of RGB and predicted depth map for an end-to-end pixel labeling.

3.1. Framework overview

Fig. 1 gives the summary of our RGBD semantic segmentation framework. As shown in the figure, the model learning part contains two stages: depth estimation from RGB images and semantic segmentation from RGB and estimated depth images. Similarly, during the application of a new RGB image, its corresponding depth image is first predicted based on the trained model and then combined with the original RGB image to further produce a detailed label map. Our main motivation for presenting this framework is based on the common observation that features extracted from the depth images can be used to facilitate the performance of many computer vision tasks. Besides that, it is also necessary to justify the performance gap between using the predicted depth and ground truth depth images.

3.2. CNN based deep-regressed depth model

To train the non-linear mapping between the RGB images and their depth maps, we adopt the deep network proposed by Ivanecky [14] which is based on the CNN model. Generally, this model first learns the global depth and local gradient information by building the global depth network and local gradient network separately. Then, based on the gradient context information and original RGB images, a refining network is learned to produce more locally-detailed depth maps updated from the previous estimated global depth map. Fig. 2 shows the general structure of deep-regressed depth estimator.

3.2.1. Learning independent deep networks for depth and gradient estimation

3.2.1.1. Global depth estimator. To estimate the depth information from RGB images, the network is trained to model the non-linear mapping between the input of RGB images and the output of ground truth depth images. The overall network structure basically follows the popular AlexNet [15], which is composed of 5 convolu-

Table 2
Evaluation metrics for depth estimation and semantic segmentation.

Depth estimation		Semantic segmentation	
RE	$\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\frac{\lvert P_i-G_i\rvert}{G_i}$	OA	$\sum_i n_{i,i}/\sum_i t_i$
SRE	$\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\frac{\lvert P_i-G_i\rvert^2}{G_i}$	MA	$\frac{1}{c}\sum_i n_{i,i}/t_i$
RMSE	$\sqrt{\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\lvert P_i-G_i\rvert^2}$	M-IU	$\frac{1}{c}\sum_i n_{i,i}/\left(t_i+\sum_j n_{j,i}-n_{i,i}\right)$
RMSE-LOG	$\sqrt{\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\lvert\log(P_i)-\log(G_i)\rvert^2}$	FW-IU	$\left(\sum_k t_k\right)^{-1}\sum_i t_i n_{i,i}/\left(t_i+\sum_j n_{j,i}-n_{i,i}\right)$
LOG10-E	$\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\lvert\log_{10}(P_i)-\log_{10}(G_i)\rvert$	CA	$n_{i,i}/t_i$
SC-INV	$\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\lvert\log(P_i)-\log(G_i)\rvert^2-\frac{1}{N^2}\left(\sum_{i\in\{1,\ldots,N\}}(\log(G_i)-\log(P_i))\right)^2$	/	/
MVN	$\sqrt{\frac{1}{N}\sum_{i\in\{1,\ldots,N\}}\left\lvert\frac{P_i-mean(P_i)}{std(P_i)}-\frac{G_i-mean(G_i)}{std(G_i)}\right\rvert^2}$	/	/
Threshold	% of P s.t. $\max\left(\frac{G_i}{P_i},\frac{P_i}{G_i}\right)<\delta,\ \delta\in\{1.25,\ 1.25^2,\ 1.25^3\}$	/	/

Table 3
Comparison of depth estimation on the NYUDv2 dataset for global depth network (GDN) and depth refinement network (DRN), and accuracy of gradient map from local gradient network (LGN).

	RE	SRE	RMSE	RMSE-LOG	LOG10-E	SC-INV	MVN	Threshold 1.25	Threshold 1.25^2	Threshold 1.25^3
GDN	1.439	4.075	2.140	1.367	0.245	1.407	0.648	0.344	0.559	0.705
LGN	–	–	265.402	–	–	–	1.292	–	–	–
DRN	1.571	4.171	2.019	1.335	0.238	1.270	0.639	0.355	0.577	0.720

Table 4
Comparison of semantic labeling on the NYUDv2 dataset for different FCN with either predicted or ground truth depth images.

	DRN + FCN32	GTD + FCN32	RGB + DRN + C-FCN32	RGB + GTD + C-FCN32	RGB + DRN + F-FCN32	RGB + GTD + F-FCN32	RGB + DRN + F-FCN16	RGB + GTD + F-FCN16
OA	0.379	0.583	0.623	0.624	0.620	0.648	0.627	0.667
MA	0.137	0.356	0.445	0.441	0.449	0.432	0.453	0.463
M-IU	0.085	0.251	0.316	0.316	0.316	0.325	0.321	0.348
FW-IU	0.225	0.419	0.463	0.463	0.460	0.480	0.470	0.506

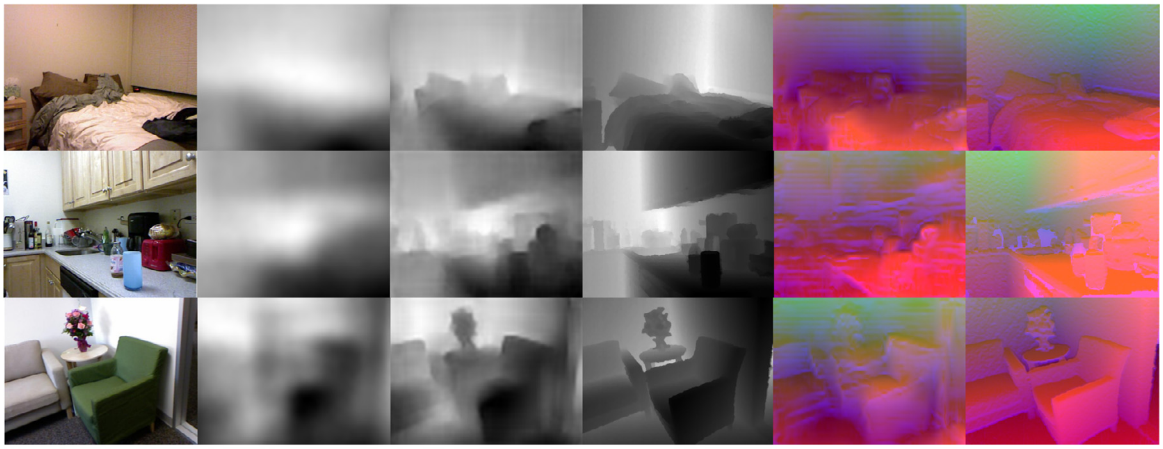

(a) RGB (b) Coarse depth of GDN (c) Refine depth of DRN (d) Ground truth depth (e) HHA for (c) (f) HHA for (d)

Fig. 6. Comparison of predicted depth and ground truth depth for NYUDv2 dataset.

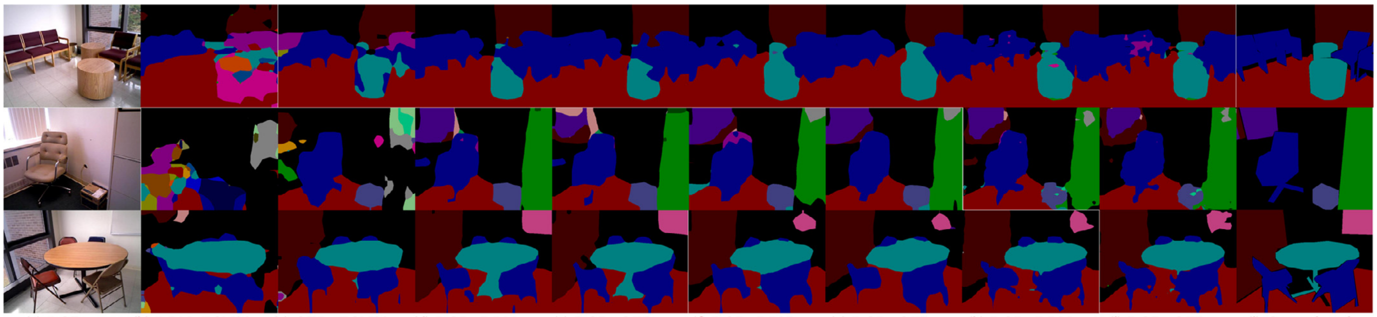
tional layers, 2 local response normalization (LRN) layers, 2 pooling layers and 2 fully connected layers. Specifically, for each convolutional layer, rectified linear units (ReLUs) is used as activation units which will trigger the learning procedure in the neuron with a positive input from some training samples. Local response normalization layer produces the response-normalized activity based on local regions and can have a benefit on generalization [15]. In this network, two LRN layers are placed after the first two convolutional layers, respectively. For the pooling layers, one is put after the first LRN layer and another is after the last convolutional layer. The usage of fully connected layers enables the learning of global context from the whole scene. The graphical illustration of the global depth network is shown in Fig. 3(a).

3.2.1.2. Local gradient estimator. Since the gradient of depth images is usually correlated with the depth changes and variation. Estimating the depth gradient can help the final refinement of depth prediction. To build the mapping between RGB images and the two-channel depth gradient maps, the deep network is trained with the similar network structure as the previous depth network without the fully connected layers. As can be seen in Fig. 3(b), the gradient network is composed of 5 convolutional layers, 1 LRN layers, and 1 pooling layers. According to the layer layout of gradient network, it is a fully convolutional network.

It is worth to be noted that a dropout layer is used in both networks for enhancing the generalization capability of the deep networks. Specifically, the dropout layer is followed by the first fully

Fig. 7. Comparison of semantic segmentation results from FCN models and ground truth for NYUDv2 dataset.

Fig. 8. Comparison of predicted depth and ground truth depth for SUNRGBD dataset.



(a) RGB (b) DRN+FCN32 (c) GTD+FCN32 (d) RGB+DRN+ C-FCN32 (e) RGB+GTD+ C-FCN32 (f) RGB+DRN+ F-FCN32 (g) RGB+GTD+ F-FCN32 (h) RGB+DRN+ F-FCN16 (i) RGB+GTD+ F-FCN16 (j) Ground truth

Fig. 9. Comparison of semantic segmentation results from FCN models and ground truth for SUNRGBD dataset.

connected layer in depth network and between the last two convolutional layers in gradient network. Besides, instead of directly using the Euclidean loss between the ground truth and the estimation to train the two models, the normalized loss function is used by first computing the mean-variance normalization of the output estimation and ground truth respectively, and then calculating their difference using Euclidean loss. In this way, the networks can be trained without the impact of the scale and translation transformation.

3.2.2. Incorporating deep depth and gradient networks for depth refinement

The accuracy of depth estimation from the above network is not good enough due to the rough prediction in lack of the depth details. Therefore, a refinement network is necessary to further increase the discrimination power based on the previously learned models. The inputs of the refining network contain three parts: the RGB images, the estimated depth and gradient maps. And the output of the refining network is an updated depth image enhanced by the context information inherently embedding in the coarse depth and gradient maps.

Fig. 3(c) gives the whole framework of the refining network. Specifically, after going through the first convolutional layer, LRN layer and pooling layer in succession, the feature maps are generated from RGB images. Then these feature maps are concatenated with the previous predicted depth and gradient based on mean-variance normalization. After the concatenation, the remaining network is composed of 4 convolutional layers and the same normalized loss function is used for optimization.

Table 5
Class-wise accuracy for 40 classes in NYUDv2 dataset.

	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	Counter	blinds	Desk	shelves	Curtain	dresser	Pillow	mirror	floor mat
DRN + FCN32	0.743	0.764	0.353	0.403	0.305	0.179	0.144	0.072	0.079	0.056	0.117	0.256	0.076	0.039	0.033	0.030	0.113	0.131	0.013	0.049
GTD + FCN32	0.842	0.929	0.663	0.789	0.690	0.639	0.429	0.116	0.487	0.363	0.233	0.687	0.158	0.156	0.091	0.094	0.309	0.531	0.372	0.041
RGB + DRN + C-FCN32	0.849	0.904	0.680	0.693	0.611	0.591	0.426	0.385	0.577	0.498	0.683	0.561	0.656	0.125	0.104	0.493	0.352	0.526	0.170	0.331
RGB + GTD + C-FCN32	0.851	0.904	0.688	0.693	0.633	0.590	0.419	0.356	0.579	0.477	0.666	0.556	0.667	0.127	0.128	0.486	0.326	0.499	0.161	0.313
RGB + DRN + F-FCN32	0.841	0.899	0.669	0.684	0.614	0.581	0.414	0.370	0.585	0.503	0.665	0.539	0.640	0.169	0.101	0.510	0.341	0.522	0.171	0.348
RGB + GTD + F-FCN32	0.904	0.954	0.696	0.814	0.718	0.691	0.506	0.223	0.594	0.475	0.464	0.714	0.481	0.151	0.139	0.268	0.381	0.576	0.304	0.080
RGB + DRN + F-FCN16	0.854	0.902	0.656	0.657	0.643	0.619	0.421	0.374	0.598	0.583	0.690	0.577	0.644	0.180	0.096	0.535	0.344	0.488	0.169	0.370
RGB + GTD + F-FCN16	0.913	0.964	0.703	0.827	0.722	0.677	0.518	0.259	0.606	0.489	0.590	0.774	0.572	0.152	0.139	0.424	0.421	0.612	0.309	0.187
	clothes	ceiling	books	Refridge rator	television	paper	towel	Shower curtain	box	Whiteboard	person	Night stand	toilet	sink	lamp	bathtub	bag	Other structure	Other furniture	Other prop
DRN + FCN32	0.046	0.425	0.022	0.010	0.094	0.019	0.027	0.033	0.005	0.035	0.016	0.033	0.328	0.097	0.017	0.011	0	0.015	0.032	0.257
GTD + FCN32	0.271	0.750	0.154	0.078	0.480	0.046	0.296	0.087	0.041	0.018	0.397	0.340	0.783	0.474	0.362	0.400	0.032	0.102	0.088	0.411
RGB + DRN + C-FCN32	0.341	0.597	0.360	0.351	0.728	0.236	0.328	0.334	0.090	0.267	0.627	0.291	0.815	0.572	0.382	0.393	0.089	0.228	0.116	0.455
RGB + GTD + C-FCN32	0.330	0.589	0.369	0.348	0.704	0.246	0.302	0.365	0.088	0.255	0.610	0.310	0.815	0.581	0.371	0.375	0.063	0.224	0.127	0.463
RGB + DRN + F-FCN32	0.363	0.595	0.395	0.379	0.660	0.292	0.331	0.349	0.115	0.246	0.622	0.346	0.807	0.580	0.406	0.406	0.104	0.215	0.115	0.454
RGB + GTD + F-FCN32	0.329	0.684	0.263	0.237	0.683	0.165	0.388	0.168	0.076	0.099	0.561	0.400	0.838	0.546	0.453	0.472	0.021	0.148	0.103	0.502
RGB + DRN + F-FCN16	0.350	0.610	0.274	0.358	0.653	0.314	0.342	0.331	0.124	0.269	0.595	0.332	0.820	0.570	0.478	0.406	0.120	0.223	0.106	0.463
RGB + GTD + F-FCN16	0.355	0.703	0.352	0.274	0.694	0.241	0.405	0.215	0.0990	0.150	0.548	0.385	0.829	0.573	0.506	0.517	0.0380	0.175	0.101	0.510

Table 6
Comparison of depth estimation on the SUNRGBD dataset for global depth network (GDN) and depth refinement network (DRN) and depth refinement network (DRN), and accuracy of gradient map from local gradient network (LGN).

	RE	SRE	RMSE	RMSE-LOG	LOG10-E	SC-INV	MVN	Threshold 1.25	Threshold 1.25^2	Threshold 1.25^3
GDN	27.775	202.003	2.391	1.732	0.308	2.625	0.805	0.292	0.508	0.662
LGN	–	–	3224.053	–	–	–	1.338	–	–	–
DRN	27.100	186.985	2.269	1.661	0.294	2.309	0.793	0.311	0.531	0.682

3.3. FCN based RGBD semantic segmentation model

Classification based on neural network is widely used for labeling each image as one specific category [23,25]. Different from that, semantic segmentation needs to generate pixel-level labeling or classification for a scene. Recently, scene labeling with the help of depth information has gained much attention. Due to the complementary information contained in depth images, RGBD data can be explored to represent more complicated and discriminative features for the labeling of different scenes. In this section, we employ two different ways of fusing RGB and depth data within the FCN framework which is proposed by Long et al. [19].

3.3.1. Channel-wise concatenation for learning FCN

With either the ground truth or the estimated depth images, the RGB images can be extended to be the four-channel images. Then the augmented data is fed into the classic FCN for the end-to-end semantic segmentation. For dense label estimation of all pixels in the input images, FCN is built according to the scheme shown in Fig. 4(a). Generally, to train an FCN between the input images and output label maps, the training dataset can include the images with different sizes and output is the label prediction with the corresponding spatial dimensions. Specifically, FCN contains 15 convolutional layers, 5 pooling layers, and 1 deconvolution layer. The first two pooling layers are placed every two convolutional layers. The last three pooling layers are located every three convolutional layers. For the last convolutional layers, it can output a predicted score vector with the dimension of class number for each coarse location. To obtain the prediction results for all images pixels, the deconvolution layer is needed to perform bilinear upsample from the coarse prediction to pixel-level prediction. Finally, softmax with loss layer is adopted to compute the logistic loss between the dense prediction scores and ground truth label map.

3.3.2. Feature-wise concatenation for learning FCN

In the above network which simply combined two image modalities can be misleading during the training since the distribution of RGB and depth images are different from each other and those difference between the depth and RGB channels is ignored. Therefore, it is meaningful to build the FCN for each modality and then concatenate their feature maps for joint prediction. The detailed model construction is shown in Fig. 4(b). Different from directly using the raw depth information for learning the deep network, HHA images are used to encode the one-channel depth images into three-channel ones, which impose the commentary inconsistency of depth distribution embedding the geocentric pose characteristic such as depth, surface normal and height. After the HHA representation, the same FCN can be constructed for the RGB and HHA data respectively. For each independent network, its structure is basically the same as shown in Fig. 4(a). Based on the coarse prediction results of the two networks, the prediction scores are first fused element-wisely and then fed into the deconvolution layer for producing the dense prediction. Softmax with loss layer is used for optimization. To further produce finer predicted depth map, responses from the forth pooling layer and deconvolution layer are fused for each network respectively, and then the predictions from two networks are finally combined as shown in Fig. 5. Since the additional prediction is from pool4 layer at stride 16, the upgraded network is named as FCN16.

4. Experiments

In this section, we evaluate the performance of depth estimation and semantic segmentation methods on two public datasets, i.e., NYUDv2 and SUBRGBD.

4.1. Experimental settings

4.1.1. Dataset preparation

Although there are multiple datasets can be used for depth estimation or semantic segmentation, few of them can be used to fulfill these two tasks simultaneously. For this reason, we choose NYUDv2 and SUNRGBD datasets, which are all indoor scenes containing both the depth and label images for the RGB frames. For the NYUDv2 dataset, it includes huge raw data (407,024 images) without labeling from Kinect and 1449 labeling data which is a subset of the raw dataset. Specifically, we first extract and preprocess 43,624 pairs of RGB and depth images from the raw dataset according to [14]. It worth noting that the related images with 1449 labeling dataset are avoided to be selected as the training dataset. Then the training dataset is augmented 10 times by scale, rotation, translation, flips, HSV shift, and change of contrast. For the labeled NYUDv2 dataset, 654 of them is used for training and the rest 795 is for training. The label images contain 40 different scene classes. As for the SUNRGBD dataset, it consists of 10,355 labeled RGBD images from different sources such as Kinect, Intel, NYUDv2, B3DO, and SUN3D. 5050 images from SUNRGBD is used for testing and the remaining 5285 is for training. The label images contain 37 different valid classes.

4.1.2. Network and hardware parameters

For all the networks, stochastic gradient descent is used for optimization. The other parameters, such as learning rate, momentum weight, decay weight for dropout and batch size, are organized in Table 1. All deep network is trained on the Caffe platform with a single NVIDIA GeForce GTX TITAN X GPU with 12GB memory.

4.1.3. Evaluation measurements

To evaluate the quality of depth estimation, the left panel of metrics in Table 2 are used, where P and G indicate the predicted depth and ground truth, respectively. N indicates the total pixel number. For the threshold measurement, higher value means better performance. For the other five measurements, relative error (RE), square relative error (SRE), root mean square error (RMSE), root mean square error log (RMSE-LOG), $\log_{10}$ error (LOG10-E), scale-invariant mean squared error (SC-INV) and mean-variance normalized error (MVN), lower value means better performance.

To evaluate the quality of semantic segmentation, the right panel of metrics in Table 2 are used, which are overall accuracy (OA), mean accuracy (MA), mean IU (M-IU) and frequency weighted IU (FW-IU) and class-wise accuracy (CA). Here, $n_{i,j}$ is the number of pixels for class i which is predicted as class j. c is the total class number. $t_i = \sum_j n_{i,j}$ is the total number of pixels belonged to class *i*. For all the metrics, higher values indicate more accurate segmentations compared to the ground truth labeling.

4.2. Evaluation on NYUDv2 dataset

Table 3 gives the comparison of depth estimation on the NYUDv2 dataset for global depth network (GDN) and depth refinement network (DRN), and accuracy of gradient map from local gradient network (LGN). It can be seen that after refinement using the gradient information, the performance of depth estimation can be improved. Table 4 shows the comparison of semantic labeling on the NYUDv2 dataset for fully convolutional network (FCN32) with either estimated depth from DRN or ground truth depth (GTD), channel-wised FCN (C-FCN32) with both RGB and depth images either from DRN or GTD, feature-wised FCN (F-FCN32 or F-FCN16) with both RGB and depth images either from DRN or GTD. Based on the experimental results, it can be seen that 1) using only the predicted depth information to train the FCN32 is basically failed for the semantic segmentation task; 2) if combing the depth with RGB images, the C-FCN32 can achieve comparable results based on either predicted depth or ground truth, which demonstrates the effectiveness of depth estimation task. However, when using ground truth depth images instead of the predicted depth, the performance of F-FCN32 increases a little; 3) by adopting the FCN16 network instead of the FCN32, more accurate segmentation can be achieved for both channel-wise and feature-wise model frameworks. To further compare the performance in respect to different classes, Table 5 lists the class-wise accuracy of the 40 classes for all the comparison methods.

For visual comparison, Fig. 6 shows the typical predicted depth images and their corresponding HHA images. Fig. 7 gives the semantic labeling results for all comparison methods indicated in Tables 3 and 4 respectively.

4.3. Evaluation on SUNRGBD dataset

Table 6 gives the comparison of depth estimation on the SUNRGBD dataset for GDN and DRN. It can be seen that similar to the results of NYUDv2, after refinement using the gradient information, the performance of depth estimation can be improved. Table 7 shows the comparison of semantic labeling on the SUNRGBD dataset for FCN32 with either estimated depth from DRN or GTD, C-FCN32 with both RGB and depth images either from DRN or GTD, F-FCN32 and F-FCN16 with both RGB and depth images either from DRN or GTD. Based on the experimental results, it can be seen that 1) if using only the predicted depth information to train the FCN32, the labeling results are much inferior to the ground truth depth. The same finding is also applied to the NYUDv2 dataset; 2) after combing the depth with RGB images, C-FCN32 with estimated depth achieves comparable performance as F-FCN16 with the ground truth depth. This finding supports the effectiveness of depth estimation method; 3) different from the results of NYUDv2 dataset, FCN16 is not always better than the FCN32 model. Table 8 lists the class-wise accuracy of the 37 classes for all the comparison methods.

Furthermore, Figs. 8 and 9 show the predicted depth images and semantic labeling results for all comparison methods indicated in Tables 6 and 7 respectively. Based on the qualitative results, it can be seen that FCN16 model can preserve much detailed boundary information compared to the FCN32.

5. Conclusion

In this paper, we investigate two important computer vision tasks both related to depth information: depth estimation and

Table 7
Comparison of semantic labeling on the SUNRGBD dataset for different FCN with either predicted or ground truth depth images.

	DRN + FCN32	GTD + FCN32	RGB + DRN + C-FCN32	RGB + GTD + C-FCN32	RGB + DRN + F-FCN32	RGB + GTD + F-FCN32	RGB + DRN + F-FCN16	RGB + GTD + F-FCN16
OA	0.505	0.689	0.732	0.732	0.710	0.728	0.714	0.734
MA	0.127	0.347	0.467	0.463	0.447	0.458	0.442	0.457
M-IU	0.084	0.251	0.340	0.340	0.323	0.334	0.322	0.337
FW-IU	0.343	0.543	0.594	0.593	0.569	0.593	0.573	0.600

Table 8
Class-wise accuracy for 37 classes in SUNRGBD dataset.

	Wall	Floor	Cabinet	Bed	Chair	Sofa	Table	Door	Window	Bookshelf	Picture	Counter	Blinds	Desk	Shelves	Curtain	Dresser	Pillow	mirror
DRN + FCN32	0.766	0.780	0.166	0.292	0.480	0.165	0.340	0.089	0.134	0.059	0.113	0.092	0.037	0.045	0.022	0.042	0.042	0.158	0.030
GTD + FCN32	0.840	0.929	0.480	0.704	0.741	0.547	0.620	0.195	0.430	0.207	0.251	0.331	0.095	0.139	0.131	0.374	0.212	0.528	0.185
RGB + DRN + C-FCN32	0.861	0.912	0.574	0.695	0.778	0.599	0.628	0.401	0.655	0.390	0.578	0.360	0.412	0.189	0.132	0.660	0.447	0.552	0.324
RGB + GTD + C-FCN32	0.866	0.901	0.560	0.699	0.801	0.576	0.630	0.406	0.637	0.380	0.555	0.355	0.407	0.195	0.139	0.655	0.451	0.543	0.295
RGB + DRN + F-FCN32	0.827	0.893	0.536	0.664	0.771	0.553	0.611	0.400	0.608	0.418	0.554	0.362	0.362	0.210	0.120	0.660	0.433	0.442	0.246
RGB + GTD + F-FCN32	0.837	0.920	0.539	0.701	0.790	0.556	0.634	0.428	0.629	0.418	0.554	0.380	0.368	0.230	0.125	0.660	0.438	0.456	0.290
RGB + DRN + F-FCN16	0.821	0.908	0.568	0.666	0.780	0.530	0.630	0.371	0.648	0.430	0.578	0.337	0.346	0.160	0.119	0.673	0.403	0.465	0.280
RGB + GTD + F-FCN16	0.841	0.937	0.578	0.703	0.789	0.567	0.656	0.363	0.664	0.435	0.589	0.363	0.354	0.171	0.122	0.672	0.417	0.461	0.313

	Floor mat	Clothes	Ceiling	Books	Fridge	Tv	Paper	Towel	Shower curtain	Box	White Board	Person	Night stand	Toilet	Sink	Lamp	Bathtub	bag
DRN + FCN32	0	0.022	0.228	0.055	0.027	0.023	0.035	0.001	0	0.018	0.032	0.008	0.025	0.201	0.081	0.025	0.033	0.019
GTD + FCN32	0	0.291	0.543	0.312	0.222	0.268	0.071	0.208	0.086	0.161	0.241	0.144	0.102	0.718	0.563	0.317	0.486	0.185
RGB + DRN + C-FCN32	0.005	0.321	0.648	0.454	0.324	0.583	0.337	0.258	0.007	0.332	0.601	0.442	0.258	0.724	0.655	0.397	0.526	0.257
RGB + GTD + C-FCN32	0	0.303	0.664	0.454	0.346	0.554	0.325	0.265	0.014	0.289	0.601	0.401	0.297	0.720	0.662	0.401	0.516	0.277
RGB + DRN + F-FCN32	0.003	0.354	0.634	0.486	0.360	0.582	0.252	0.279	0.024	0.280	0.516	0.463	0.187	0.722	0.621	0.317	0.545	0.241
RGB + GTD + F-FCN32	0.004	0.348	0.660	0.490	0.367	0.603	0.244	0.290	0.033	0.288	0.507	0.477	0.185	0.726	0.650	0.314	0.556	0.252
RGB + DRN + F-FCN16	0	0.345	0.661	0.490	0.318	0.553	0.235	0.257	0.009	0.286	0.524	0.459	0.108	0.686	0.615	0.296	0.529	0.277
RGB + GTD + F-FCN16	0.003	0.349	0.687	0.504	0.326	0.567	0.240	0.284	0.0150	0.290	0.547	0.463	0.110	0.684	0.642	0.337	0.554	0.300

RGBD semantic segmentation. For the first task, a deep depth repressor is realized by first the joint learning the depth and its gradient and then refining the depth based on the complementary context information from obtained coarse results. For the second task, two deep models are used to train the FCNs on RGBD data, which use the raw depth images and their further encoding by HHA representation, respectively. By combing two tasks into one framework, the predicted depth can be used to assist the semantic segmentation of raw RGB images, which is a benefit to those datasets in the lack of the corresponding depth information. Experimental results on the NYUDv2 and SUNRGBD dataset demonstrate the effectiveness of the unified framework. In the future, we will further improve the network design based on the multi-task learning strategy. In this way, the performance of the two tasks is expected to be improved since their relationship can be better explored to help the learning of each other.

References

[1] A. Bansal, X. Chen, B. Russell, A. Gupta, D. Ramanan. PixelNet: representation of the pixels, by the pixels, and for the pixels. CoRR abs/1702.06506 (2017).

[2] Y. Cao, C. Shen, H.T. Shen, Exploiting depth from single monocular images for object detection and semantic segmentation, IEEE Trans. Image Process. 26 (2017) 836–846.

[3] L. Chi, J. Bohren, E. Carlson, G.D. Hager, Hierarchical semantic parsing for object pose estimation in densely cluttered scenes, in: 2016 IEEE International Conference on Robotics and Automation (ICRA), 2016, pp. 5068–5075.

[4] P. Dong, L. Wang, W. Lin, D. Shen, G. Wu, Scalable joint segmentation and registration framework for infant brain images, Neurocomputing 229 (2017) 54–62.

[5] S. Du, Y. Guo, G. Sanroma, D. Ni, G. Wu, D. Shen, Building dynamic population graph for accurate correspondence detection, Med. Image Anal. 26 (2015) 256–267.

[6] S. Du, J. Liu, C. Zhang, J. Zhu, K. Li, Probability iterative closest point algorithm for m-D point set registration with noise, Neurocomputing 157 (2015) 187–198.

[7] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2650–2658.

[8] D. Eigen, C. Puhrsch, R. Fergus, Depth map prediction from a single image using a multi-scale deep network, in: Proceedings of the 27th International Conference on Neural Information Processing Systems, MIT Press, Montreal, Canada, 2014, pp. 2366–2374.

[9] [9] R. Garg, B.G.V. Kumar, I.D. Reid. Unsupervised CNN for single view depth estimation: geometry to the rescue. CoRR abs/1603.04992 (2016).

[10] C. Godard, O.M. Aodha, G.J. Brostow. Unsupervised monocular depth estimation with left-right consistency. CoRR abs/1609.03677 (2016).

[11] S. Gupta, P. Arbeláez, J. Malik, Perceptual organization and recognition of indoor scenes from RGB-D images, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 564–571.

[12] S. Hao, M. Wang, R. Hong, J. Jiang, Spatially guided local laplacian filter for nature image detail enhancement, Multimedia Tools Appl. 75 (2016) 1529–1542.

[13] R. Hong, M. Wang, Y. Gao, D. Tao, X. Li, X. Wu, Image annotation by multiple-instance learning with discriminative feature mapping and selection, IEEE Trans. Cybern. 44 (2014) 669–680.

[14] J. Ivanecky, Depth Estimation by Convolutional Neural Networks, Brno University of Technology, 2016.

[15] A. Krizhevsky, I. Sutskever, G. Hinton, ImageNet classification with deep convolutional neural networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[16] Z. Li, Y. Gan, X. Liang, Y. Yu, H. Cheng, L. Lin, LSTM-CF: unifying context modeling and fusion with LSTMs for RGB-D scene labeling, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II, Springer International Publishing, Cham, 2016, pp. 541–557.

[17] B. Liu, S. Gould, D. Koller, Single image depth estimation from predicted semantic labels, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 1253–1260.

[18] F. Liu, C. Shen, G. Lin, I. Reid, Learning depth from single monocular images using deep convolutional neural fields, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2016) 2024–2039.

[19] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431–3440.

[20] W. Peng, S. Xiaohui, L. Zhe, S. Cohen, B. Price, A. Yuille, Towards unified depth and semantic prediction from a single image, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 2800–2809.

[21] A. Roy, S. Todorovic, Monocular depth estimation using neural regression forest, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 5506–5514.

[22] A. Saxena, S.H. Chung, A.Y. Ng, Learning depth from single monocular images, in: Proceedings of the 18th International Conference on Neural Information Processing Systems, MIT Press, Vancouver, British Columbia, Canada, 2005, pp. 1161–1168.

[23] J. Shi, J. Wu, Y. Li, Q. Zhang, S. Ying, Histopathological image classification with color pattern random binary hashing based PCANet and matrix-form classifier, IEEE J. Biomed. Health Inform. (2017) 1–1.

[24] J. Shi, X. Zheng, Y. Li, Q. Zhang, S. Ying, Multimodal neuroimaging feature learning with multimodal stacked deep polynomial networks for diagnosis of alzheimer's disease, IEEE J. Biomed. Health Inform. (2017) 1–1.

[25] J. Shi, S. Zhou, X. Liu, Q. Zhang, M. Lu, T. Wang, Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset, Neurocomputing 194 (2016) 87–94.

[26] M. Wang, W. Fu, S. Hao, H. Liu, X. Wu, Learning on big graph: label inference and regularization with anchor hierarchy, IEEE Trans. Knowl. Data Eng. 29 (2017) 1101–1114.

[27] D. Wolf, J. Prankl, M. Vincze, Enhancing semantic segmentation for robotics: the power of 3-D entangled forests, IEEE Robot. Autom. Lett. 1 (2016) 49–56.

[28] Z. Wu, X. Jiang, N. Zheng, Y. Liu, D. Cheng, Exact solution to median surface problem using 3D graph search and application to parameter space exploration, Pattern Recognit. 48 (2015) 380–390.

[29] S. Ying, Z. Wen, J. Shi, Y. Peng, J. Peng, H. Qiao, Manifold preserving: an intrinsic approach for semisupervised distance metric learning, IEEE Trans. Neural Netw. Learn. Syst. (2017) 1–12.

[30] S. Ying, G. Wu, Q. Wang, D. Shen, Hierarchical unbiased graph shrinkage (HUGS): a novel groupwise registration for large data set, NeuroImage 84 (2014) 626–638.

[31] X. Zhu, Z. Huang, H.T. Shen, X. Zhao, Linear cross-modal hashing for efficient multimedia search, in: Proceedings of the 21st ACM International Conference on Multimedia, ACM, Barcelona, Spain, 2013, pp. 143–152.

[32] X. Zhu, X. Li, S. Zhang, Block-row sparse multiview multilabel learning for image classification, IEEE Trans. Cybern. 46 (2016) 450–461.

[33] X. Zhu, X. Li, S. Zhang, C. Ju, X. Wu, Robust joint graph sparse coding for unsupervised spectral feature selection, IEEE Trans. Neural Netw. Learn. Syst. 28 (2017) 1263–1275.

[34] X. Zhu, L. Zhang, Z. Huang, A sparse embedding and least variance encoding approach to hashing, IEEE Trans. Image Process. 23 (2014) 3737–3750.